1.  阿里一面（电面，45min）
   1. 自我介绍（学校学习课程和项目）
   2. 看过哪些书？说了一堆（西瓜书，李航之类的）。说完，面试官疑惑的问：你没看过PRML？黑人问号，赶紧把吴恩达视频和cs231n拿出来压压惊（后来实习才知道，面试官正在看PRML这本书）。
   3. 说了一下大致的项目，大概介绍一下（面试官评价基础很扎实）
   4. 问了loss优化方法，说了BGD，SGD，各自优缺点，优化方向（Adam之类的）
   5. 问了一个开放题，说是考验一下我的反应能力（阿里确实比较喜欢脑子灵活的同学）。题目：用户打开一个App时，我们可以得到用户的坐标（经纬度），那么如何根据经纬度得到城市名称呢？回答：没有预先定义的数据库，智能调用高德等第三方接口（因为高德被阿里收购了嘛）；不要求做高精度定位，可以将大城市为中心构建区域块（所有区域块内的经纬度映射到这个城市。**这种题目要紧密集合业务来说**）
2. 阿里二面（电面，50min）
   1. 自我介绍
   2. 项目问问
   3. 你了解决策树吗？回答：ID3 C4.5 优缺点 树的融合（GBDT，RF） 我的实现。这里注意，面试官只问“你了解决策树吗？”，我的回答比较丰富，**这里面试的一个tips就是，要尽可能主动的向面试官灌输你会什么内容，做好知识输出。**
   4. L1 L2了解吗？回答：L1 L2的作用，为什么有这样的作用？一般求L1的优化方法（坐标下降，LARS角回归）（面试官一脸懵逼，你们老师这都讲吗？我说我是自己看的。。。深藏功与名）
   5. 链表逆序你会吗？回答：非递归 递归
   6. 开放题，如何判断一个query是时效性query（答得比较差，就不贴了）。为什么面算法会有这种问题？因为面试官原来是做搜索的，专门搞时效性query，mmp。
3. 阿里HR面（视频，10min）
   1. 自我介绍，讲一下你的优势，你对我们团队了解吗？blabla，10分钟搞定。
4. 腾讯OMG内推一面（视频，60min）
   1. 上来就问项目，从头说到尾
   2. 说一下LeNet的网络结构，一层层说，带着卷积核大小，越详细越好
   3. 项目里采集数据的流程
   4. 问kaggle比赛的项目（竟然不知道kaggle是啥，醉了）
   5. kaggle项目里如何处理数据，计算特征相关性用什么办法
   6. 缺失值怎么处理？验证集怎么划分？哪些指标说明你的模型调优了？调节过模型的哪些参数
   7. 过拟合的标志有哪些？
   8. 决策树如何后剪枝
   9. 编程题，2sum（恩，面试官不同C 11语法）
5. 腾讯OMG内推二面（视频，30分钟）
   1. 妹的，8点打电话，让我准备环境，8点半视频面试，真着急
   2. 上来编程（然而一些特殊情况没有考虑到，在面试官的提醒下也没做出来，我就知道内推凉了，怪我菜，认栽）
   3. 什么是Kmeans，与EM怎么联系
   4. 介绍下决策树，说一下属性选择方法
   5. 说下PCA
   6. 第一个编程题你会了吗（mmp，就顾着回答你问题了，我哪有时间思考？不会，然后挂了）
6. 腾讯线下面试WXG一面（现场，90min）
   1. 因为腾讯每年都要来哈工大线下面试，所以相当于多了一次机会
   2. 面试官在处理公司的事儿，等了很长时间
   3. 上来给了一页题目，指了两个，你先做着吧，我这还有点事儿。一道题目是智力题，64匹马，8个赛道，找出最快的4匹马。编程题是类似于一个归并排序的东西。（**写面经的时候，被实验室ACM大佬看到了，甩了我一句，你考虑过4个赛道怎么解决吗？给跪了**）
   4. 项目说一遍，老生常谈
   5. BP神经网络推导一遍
   6. 其他细节记不清了，反正都是基础就对了
   7. 最后补了一句，你要是会点NLP知识就好了
7. 腾讯线下面试WXG二面（现场，40min）
   1. 进了房间，跟面试官对视了十几秒，面试官开口了：你怎么这么被动，要回推销自己，ok？
   2. 然后我就开始了我得表演，从决策树到SVM，从BP到CNN，基本上把我会的都喷了。
   3. 面试官感觉我刹不住车了，然后就叫停了。问了一道题，随机数1~5，如何生成随机数1~7
8. 腾讯线下面试WXG  HR面（现场 ，20分钟）
   1. 你对机器学习的理解
   2. 除了CNN还熟悉其他深度学习模型吗？
   3. 学校情况（导师是谁，腾讯的HR很关心你的导师是谁？）
   4. 你有什么想问的？
   5. 最终，经过煎熬的等待，给了劳务实习生offer（**劳务这种，比较坑人，幸好没去，类似外包，建议大家也不要去**）
9. 美团面试
   1. 阴差阳错，师兄给投成了“应届生春招找工作”这种情况，经过多方沟通无果后，没能得到美团的面试机会。
10. 今日头条
    1. 笔试挂
11. 滴滴
    1. 二面挂（可能是方向不太符合，一直问题python和C 底层原理的内容）
12. 搜狗
    1. 投的太晚了，没有面试机会

 秋招过程

1. 秋招投了腾讯（提前批），今日头条，百度，美团，京东，搜狗，freewheel，Amazon
2. 本来在阿里实习，转正答辩通过，可以拿到offer，但是base地在杭州，与我的意向不是很符合，所以放弃了。不过在此感谢实习期间所有的阿里师兄对我的帮助，阿里的项目经历也成为我秋招面试时的宝贵财富。
3. 腾讯MIG一面（电面，40min）
   1. 只说了阿里的项目，就结束了。
4. 腾讯MIG二面（电面，40min）
   1. 阿里的项目，然后聊聊对推荐的认识，谈话内容主要集中在推荐领域
   2. 编程题：怎么判断链表是否有环，还没说完，面试官就说不用说了，你有啥想问的，哈哈
5. 腾讯MIG三面（电面，50min）
   1. 这一面，比较困难，因为面试官完全不看简历，完全不用你做自我介绍，上来直接开始怼
   2. SVM与LR的优缺点（竟然还有人问这种问题）
   3. SVM与LR的应用场景，那么更适用于这种场景？
   4. GBDT，RF，XGboost相关
   5. 特征提取方法，如何判断特征是否重要
   6. 如何采集样本，样本类别不均衡对模型有什么影响（中插一条消息：很多伙伴私信我要微信号，当然我很愿意与大家交流。秋招过去了，我也该忙别的事了，可能最近都不来牛客了，我的微信是wang9448664  耐心的你会看见的）（此时，面试官一度怀疑简历上的项目是否是真的）
   7. 其他的一些细节问题，在你的回答里挑问题，不是主动问的（**所以TIPs是：可以通过自己的回答来引导面试官的面试方向，但是要有度，否则容易引火烧身**）
   8. 编程题目：有序数组的交集（这个算法主要是搜索领域经常用，如果能给出在搜素领域的优化方式，那就能拿下这个面试官了。可惜太菜，我只给到了O（n）的时间复杂度）
   9. 最后补了一句：前面答的还不错
6. 腾讯MIG四面（视频，30min）
   1. 这一面应该是技术总监，不会怼人的那种
   2. 基本就是聊聊项目，最近看什么论文？（**是否关注论文这点很重要，反应你对前言技术的关注度**）
   3. 技术的大方向，不会问具体的细节
   4. 所以比较轻松
7. 腾讯MIG HR面（电面，20min）
   1. 了解你本科学校，研究生导师（腾讯HR真的很关心导师是谁）
   2. 为什么选择推荐方向
   3. 实验室的工作，扮演的角色是什么？
   4. 你有什么优缺点，
   5. 确定意向
   6. 另外提一句，9月11二三面，9月12四面，HR面，说是为了在笔试前走完流程，直接累的我腰疼。
8. 今日头条
   1. 还在实习的时候，今日头条就开始提前批了（需要白金码）。本来有一个白金码但是因为还没准备好，所以没投
   2. 后来面试的同学都拿到了offer，就慌了，于是匆忙的内推了一份
   3. 笔试挂（**两次笔试都挂了，足以证明我就是个菜鸟**）
9. 美团
   1. 一面讲了项目。然后面试官开始问你会不会这个东西，你只需要回答会或者不会，不需要具体讲解（可能面试官比较信任吧）
   2. 后来因为拿到了腾讯的offer，所以就推掉了之后的面试（面试官还问美团匹配和腾讯一样的岗位，你考虑吗?哈哈，这样的感觉真好）
10. 搜狗
    1. 一面，项目 编程题（数组中第一个大于等于K的数，判断树是否相等（同构 对应节点值相等））
    2. HR面也推掉了（很抱歉搜狗了，那边邀约了很多次，一直让我考虑搜狗。搜狗也很优秀，但是还是更喜欢腾讯的岗位，只能抱歉了）
11. Amazon
    1. 外企特点是连续两面，绝不拖泥带水
    2. 一面主要聊项目。面试官比较慈祥。因为二面面试官还在面试，之后聊了一些学校的事儿。期间一直询问是否愿意做软件开发（果断不妥协）
    3. 二面，聊了一下项目。然后问了一些基础东西（C 虚函数 原理，实现ls功能，LCA（树中两个节点的最低公共祖先））
    4. 目前还没结果
12. FreeWheel
    1. 之前没听说过这家公司，不过据说是一家不错的外企
    2. 一面，主要说项目，然后英语“谈笑风生”了一段
    3. 二面，又说了一遍项目。编程题（有序数组的交集，是的，和腾讯问的一样），然后又谈笑风生了一段
    4. 目前还没结果
13. 其他的都是投了，但是还没有面试，估计不会面了。

 面试总结

1. 实习经历可以是一个金钥匙
   1. 从上边的面试经历可以看得出，春招面试还是比较痛苦的，一般时间较长，且大多数时间纠结在基础知识。所以春招的面试准备重在基础和刷题，因为大部分人都没有项目经历，所以只能靠基础知识评价你的能力了。
   2. 春招如果能拿到一个不错公司的offer，并做了一个较为完整的项目，那么恭喜你，秋招很简单。比如我的阿里实习经历，面试官看了项目以后就默认你有了较为扎实的理论基础（毕竟去BAT实习的还是少数啊），大部分考察你的实践能力，也就是问项目。
2. 简历
   1. 项目一定要真实具体。一个项目能够完整的从头到尾叙述下来，对于其中各种出现的问题，要有合理的解释。你在叙述项目的过程中，面试官会随时打断你，问你为什么？胡乱编个项目蒙混过关？不存在的。
   2. 如果是现场面试的话，一定要画结构图。因为面试官也是普通人，听你说一遍，并不能对你的项目有一个直观的认识，所以画图最好
   3. 最容易忽略的两个点：一是项目来源，项目背景；二是项目的创新点。大多数人在复述项目的时候一再强调各种花里胡哨的技术，这会另面试官反感的。阐述项目来源会让面试官更容易理解你项目的意义，否则说了半天，面试官不知道你在解决什么问题，那就囧了；其次最重要的是创新点，或者解决了哪些难点，如果一个项目很简单，或者是已经有成熟的解决方案，那你的项目意义在哪里呢？
   4. 如果你的目标是算法工程师，就不要让社团活动、优秀班干部占据太多的篇幅，一页简历空间就那么大，在有限的空间里尽可能展示自己的技术实力。
3. 我理解等面试结果的你
   1. 等面试结果是痛苦的，我经历过两次。现在牛客里流传“终于收到某某公司的短信了”，结果一看是业务推销短信。然后大家一致评论“又疯了一个”。很有意思
   2. 正视自己。不排除有非本人因素造成面试失败的，但是这绝对是极小概率事件。通常来讲，面试失败了，还是因为你某些地方还没理解透。你可能SVM推导卡在了一个步骤上，然后你抱怨说我只要看一眼书就会了，但是对不起，面试官认为你是比那些能顺利推导的同学是差一些的。

 给明年的你

1. 成为算法工程师，应该学习哪些东西
   1. 首先说算法工程师有几个方向：NLP，推荐，CV，深度学习，然后结合公司业务做得内容各不相同
   2. 传统机器学习算法：感知机，SVM，LR，softmax，Kmeans，DBSCAN，决策树（CART，ID3，C45），GBDT，RF，Adaboost，xgboost，EM，BP神经网络，朴素贝叶斯，LDA，PCA，核函数，最大熵等
   3. 深度学习：CNN，RNN，LSTM，常用激活函数，Adam等优化算法，梯度消失（爆炸）等
   4. 推荐系统：itemBasedCF，userBasedCF，冷启动，SVD（各种变形），FM，LFM等
   5. NLP：TF-IDF，textrank，word2vec(能推导，看过源码)，LCA，simhash
   6. 常见概念：最大似然估计，最小二乘法，模型融合方法，L1L2正则（Lasso，elestic net），判别式模型与生成式模型，熵-交叉熵-KL散度，数据归一化，最优化方法（梯度下降，牛顿法，共轭梯度法），无偏估计，F1（ROC，recall，precision等），交叉验证，bias-variance-tradeoff，皮尔逊系数，
   7. 概率论，高数，线性代数（像我一样懒的人，就可以遇到哪里复习哪里，:D）
   8. 常见问题（具体答案去搜知乎或者百度，最好能在实际项目中总结出来）：
      1. 常见损失函数
      2. SGD与BGD
      3. 如何处理样本非均衡问题
      4. 过拟合原因，以及解决办法
      5. 如何处理数据缺失问题
      6. 如何选择特征
      7. L1为什么能让参数稀疏，L2为什么会让参数趋于较小值，L1优化方法
      8. 各模型的优缺点，以及适用场景
   9. 学明白上述所有内容你需要多长时间？反正我这么笨的人用了不到一年时间（我本科完全没接触过算法相关，完全是研一学的）
2. 推荐书籍
   1. C ：《C primer5》《STL源码分析》《深度探索C 对象模型》《Effective C 》《Effective STL》 （虽然有些书有点老，不过开卷有益吧）（其他语言就不管了哈）
   2. python：《python学习手册》《python源码分析》《改善python程序的91个建议》（Python必须要会）
   3. 刷题：《编程之美》《剑指offer》《程序员代码面试指南》《leetcode》
   4. 算法相关：《统计学习方法》（这本多看）《数据挖掘导论》《数学之美》《田林轩视频》《吴恩达视频》《西瓜书》
3. 简历项目
   1. 最好能有两个相关的项目，而且是有质量的，不要太水
   2. 没有项目的，可以去参见比赛（kaggle，天池），比赛成绩高，比项目管用。成绩不高的，一定要有自己的解决方案。
4. 刷题
   1. 刷题是必须的，书目就是上边列的哪些
   2. 每天一道或者两道，风吹雨打也不能停。如果坚持住，一年后你就成了

 其他

1. 什么叫学透了，学明白了
   1. 别人问你问题，你能讲明白
   2. 躺在床上，闭着眼，能完整的阐述一个算法（什么是完整？以SVM为例，SVM的推导？KKT条件？什么是支持向量？什么是松弛变量？为什么推导成对偶形式？核函数的作用是什么？如何选择核函数？模型优缺点？）你说这些问题我都明白，但是你是否能形成一个知识体系呢？一提到SVM，就能想到所有这些问题呢？
   3. 能够达到第二步的要求，那么面试官在问“说一下你对SVM的认识”，你就可以滔滔不绝的讲了，这样面试官才能认可你，这样就很舒服了。
2. 人脉太重要了
   1. 找工作时，互相帮助，多加几个交流群，观摩大佬的一举一动
   2. 多和上班的师兄沟通，因为他们能把简历直接给到leader手里
   3. 多向周围的大神学习。就像我实验室的ACM大佬，他是我让我佩服的五体投地的存在。每次有什么不会做的编程题，找他解释都是秒解。还要感谢实验室的师兄，带我项目，助我去阿里实习。